

# High Speed VLSI Architecture for 3-D Discrete Wavelet Transform

B.K.N.Srinivasarao and Indrajit Chakrabarti

Department of Electronics and Electrical Communication Engineering

Indian Institute of Technology, Kharagpur, INDIA

E.Mail : srinu.bkn@iitkgp.ac.in, indrajit@ece.iitkgp.ernet.in

## Abstract

This paper presents a memory efficient, high throughput parallel lifting based running three dimensional discrete wavelet transform (3-D DWT) architecture. 3-D DWT is constructed by combining the two spatial and four temporal processors. Spatial processor (SP) apply the two dimensional DWT on a frame, using lifting based 9/7 filter bank through the row processor (RP) in row direction and then apply in the column direction through column processor (CP). To reduce the temporal memory and the latency, the temporal processor (TP) has been designed with lifting based 1-D Haar wavelet filter. The proposed architecture replaced the multiplications by pipeline shift-add operations to reduce the CPD. Two spatial processors works simultaneously on two adjacent frames and provide 2-D DWT coefficients as inputs to the temporal processors. TPs apply the one dimensional DWT in temporal direction and provide eight 3-D DWT coefficients per clock (throughput). Higher throughput reduces the computing cycles per frame and enable the lower power consumption. Implementation results shows that the proposed architecture has the advantage in reduced memory, low power consumption, low latency, and high throughput over the existing designs. The RTL of the proposed architecture is described using verilog and synthesized using 90-nm technology CMOS standard cell library and results show that it consumes 43.42 mW power and occupies an area equivalent to 231.45 K equivalent gate at frequency of 200 MHz. The proposed architecture has also been synthesised for the Xilinx zynq 7020 series field programmable gate array (FPGA).

## Index Terms

Index Terms : discrete wavelet transform, 3-D DWT, lifting based DWT, VLSI Architecture, flipping structure, strip-based scanning.

## I. INTRODUCTION

Video compression is a major requirement in many of the recent applications like medical imaging, studio applications and broadcasting applications. Compression ratio of the encoder completely depends on the underlying compression algorithms. The goal of compression techniques is to reduce the immense

amount of visual information to a manageable size so that it can be efficiently stored, transmitted, and displayed. 3-D DWT based compressing system enables the compression in spatial as well as temporal direction which is more suitable for video compression. Moreover, wavelet based compression provide the scalability with the levels of decomposition. Due to continuous increase in size of the video frames (HD to UHD), video processing through software coding tools is more complex. Dedicated hardware only can give higher performance for high resolution video processing. In this scenario there is a strong requirement to implement a VLSI architecture for efficient 3-D DWT processor, which consumes less power, area efficient, memory efficient and should operate with a higher frequency to use in real-time applications.

From the last two decades, several hardware designs have been noted for implementation of 2-D DWT and 3-D DWT for different applications. Majority of the designs are developed based on three categories, viz. (i) convolution based (ii) lifting-based and (iii) B-Spline based. Most of the existing architectures are facing the difficulty with larger memory requirement, lower throughput, and complex control circuit. In general the circuit complexity is denoted by two major components viz, arithmetic and Memory component. Arithmetic component includes adders and multipliers, whereas memory component consists of temporal memory and transpose memory. Complexity of the arithmetic components is fully depends on the DWT filter length. In contrast size of the memory component is depends on dimensions of the image. As image resolutions are continuously increasing (HD to UHD), image dimensions are very high compared to filter length of the DWT, as a result complexity of the memory component occupied major share in the overall complexity of DWT architecture.

Convolution based implementations [1]-[3] provides the outputs within less time but require high amount of arithmetic resources, memory intensive and occupy larger area to implement. Lifting based implementations requires less memory, less arithmetic complex and possibility to implement in parallel. However it require long critical path, recently huge number of contributions are noted to reduce the critical path in lifting based implementations. For a general lifting based structure [4] provides critical path of  $4T_m + 8T_a$ , by introducing 4 stage pipeline it cut down to  $T_m + 2T_a$ . In [5] Huang et al., introduced a flipping structure it further reduced the critical path to  $T_m + T_a$ . Though, it reduced the critical path delay in lifting based implementation, it requires to improve the memory efficiency. Majority of the designs

implement the 2-D DWT, first by applying 1-D DWT in row-wise and then apply 1-D DWT in column wise. It require huge amount of memory to store these intermediate coefficients. To reduce this memory requirements, several DWT architecture have been proposed by using line based scanning methods [7]-[11]. Huang et al., [7]-[8] give brief details of B-Spline based 2-D IDWT implementation and discussed the memory requirements for different scan techniques and also proposed a efficient overlapped strip-based scanning to reduce the internal memory size. Several parallel architectures were proposed for lifting-based 2-D DWT [8]-[17]. Y. Hu et al. [17], proposed a modified strip based scanning and parallel architecture for 2-D DWT is the best memory-efficient design among the existing 2-D DWT architectures, it requires only  $3N + 24P$  of on chip memory for a  $N \times N$  image with  $P$  parallel processing units (PU). Several lifting based 3-D DWT architectures are noted in the literature [18]-[24] to reduce the critical path of the 1-D DWT architecture and to decrease the memory requirement of the 3-D architecture. Among the best existing designs of 3-D DWT, Darji et al. [24] produced best results by reducing the memory requirements and gives the throughput of 4 results/cycle. Still it requires the large on-chip memory ( $4N^2 + 10N$ ).

In this paper, we propose a new parallel and memory efficient lifting based 3-D DWT architecture, requires only  $2*(3N + 60P) + 48$  words of on-chip memory and produce 8 results/cycle. The proposed 3-D DWT architecture is built with two spatial 2-D DWT (CDF 9/7) processors and four temporal 1-D DWT (Haar) processors. Proposed architecture for 3-D DWT replaced the multiplication operations by shift and add, it reduce the CPD from  $T_m + T_a$  to  $4T_a$ . Further reduction of CPD to  $T_a$  is done by introducing pipeline in the processing elements. To eliminate the temporal memory and to reduce the latency, Haar wavelet is incorporated in temporal processor. The resultant architecture has reduce the latency, on chip memory and to increase the speed of operation compared to existing 3-D DWT designs. The following sections provide the architectural details of proposed 3-D DWT through spatial and temporal processors.

Organization of the paper as follows. Theoretical background for DWT is given in section II. Detailed description of the proposed architecture for 3-D DWT is provided in section III. Implementation results and performance comparison is given in Section IV. Finally, concluding remarks are given in Section V.

## II. THEORETICAL BACKGROUND

Lifting based wavelet transform designed by using a series of matrix decomposition specified by the Daubechies and Sweledens in [4]. By applying the flipping [5] to the lifting scheme, the multipliers in the longest delay path are eliminated, resulting in a shorter critical path. The original data on which DWT is

applied is denoted by  $X[n]$ , and the 1-D DWT outputs are the detail coefficients  $H[n]$  and approximation coefficients  $L[n]$ . For the Image (2-D) above process is performed in rows and columns as well. Eqns.(1)-(6) are the design equations for flipping based lifting (9/7) 1-D DWT [6] and the same equations are used to implement the proposed row processor (1-D DWT) and column processor (1-D DWT).

$$H_1[n] \leftarrow a' * X[2n - 1] + \{X[2n] + X[2n - 2]\} \dots P1 \quad (1)$$

$$L_1[n] \leftarrow b' * X[2n] + \{H_1[n] + H_1[n - 1]\} \dots U1 \quad (2)$$

$$H_2[n] \leftarrow c' * H_1[n] + \{L_1[n] + L_1[n - 1]\} \dots P2 \quad (3)$$

$$L_2[n] \leftarrow d' * L_1[n] + \{H_2[n] + H_2[n - 1]\} \dots U2 \quad (4)$$

$$H[n] \leftarrow K0 * \{H_2[n]\} \quad (5)$$

$$L[n] \leftarrow K1 * \{L_2[n]\} \quad (6)$$

Where  $a' = 1/\alpha$ ,  $b' = 1/\alpha\beta$ ,  $c' = 1/\beta\gamma$ ,  $d' = 1/\gamma\delta$ ,  $K0 = \alpha\beta\gamma/\zeta$ , and  $K1 = \alpha\beta\gamma\delta\zeta$  [4]. The lifting step coefficients  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\delta$  and scaling coefficient  $\zeta$  are constants and its values  $\alpha = -1.586134342$ ,  $\beta = -0.052980118$ ,  $\gamma = 0.8829110762$ , and  $\delta = 0.4435068522$ , and  $\zeta = 1.149604398$ .

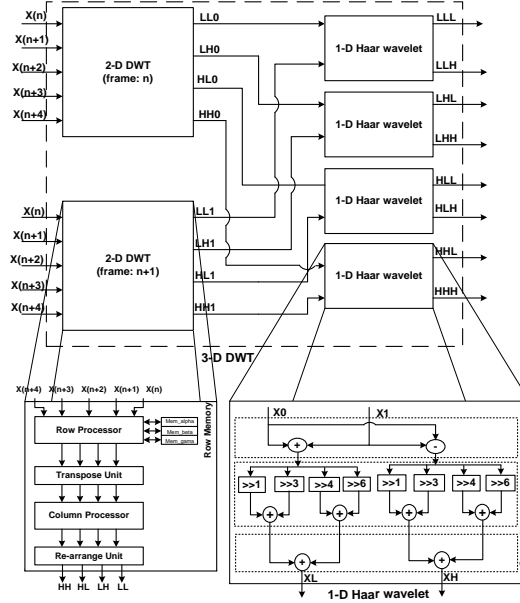
Lifting based wavelets are always memory efficient and easy to implement in hardware. The lifting scheme consists of three steps to decompose the samples, namely, splitting, predicting (eqn. (1) and (3)), and updating (eqn. (2) and (4)).

Haar wavelet transform is orthogonal and simple to construct and provide fast output. By considering the advantages of the Haar wavelets, the proposed architecture uses the Haar wavelet to perform the 1-D DWT in temporal direction (between two adjacent frames). Sweldens *et al.* [25] developed a lifting based Haar wavelet. The equations of the lifting scheme for the Haar wavelet transform is as shown in eqn.(7)

$$\begin{bmatrix} L \\ H \end{bmatrix} = \begin{pmatrix} \sqrt{2} & 0 \\ 0 & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & S(z) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -P(z) & 1 \end{pmatrix} \begin{pmatrix} X_0(z) \\ X_1(z) \end{pmatrix} \quad (7)$$

$$\begin{aligned} L &= \frac{1}{\sqrt{2}}(X_0 + X_1) \\ H &= \frac{1}{\sqrt{2}}(X_1 - X_0) \end{aligned} \quad (8)$$

Eqn.(8) is extracted by substituting Predict value  $P(z)$  as 1 and Update step  $S(z)$  value as 1/2 in eqn.(7), which is used to develop the temporal processor to apply 1-D DWT in temporal direction ( $3^{rd}$  dimension).



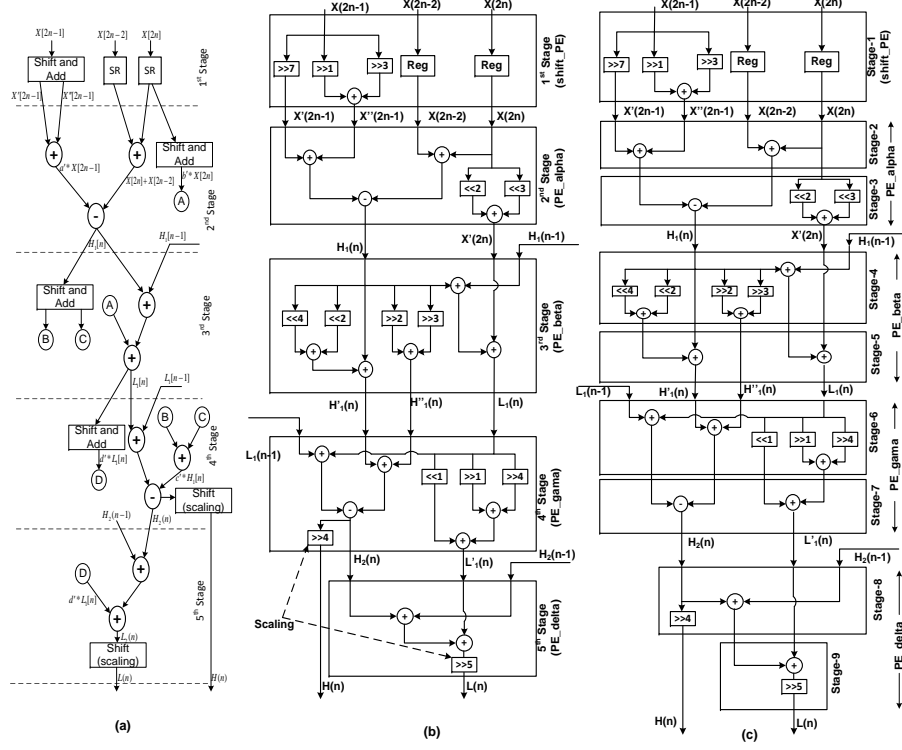


Figure 2. (a) Data Flow Graph of Processing Unit (b) Processing Unit with five pipeline stages (c) Processing Unit with nine pipeline stages

has been constructed with two pipeline stages for further reduction of CPD. This modified PU, reduces the CPD to  $T_a$  (adder delay). Fig. 1 shows that the number of inputs to the spatial processor is equal to  $2P+1$ , which is also equal to the width of the strip. Where  $P$  is the number of parallel processing units (PUs) in the row processor as well as column processor. We have designed the proposed architecture with two parallel processing units ( $P = 2$ ). The same structure can be extended to  $P = 4, 8, 16$  or  $32$  depending on external bandwidth. Whenever row processor produces the intermediate results, immediately column processor start to process on those intermediate results. Row processor takes 9 clocks to produce the temporary results then after column processor takes 9 more clocks to to give the 2-D DWT output; finally, temporal processor takes 3 more clocks after 2-D DWT results are available to produce 3-D DWT output. As a summary, proposed 2-D DWT and 3-D DWT architectures have constant latency of 18 and 21 clock cycles respectively, regardless of image size  $N$  and number of parallel PUs ( $P$ ). Details of the row processor and column processor are given in the following sub-sections.

*1) Row Processor (RP):* Let  $X$  be the image of size  $N \times N$ , extend this image by one column by using symmetric extension. Now image size is  $N \times (N + 1)$ . Refer [17] for the structure of strip based scanning method. The proposed architecture initiates the DWT process in row wise through row processor (RP) then process the column DWT by column processor (CP). Fig. 3(a). shows the generalized structure

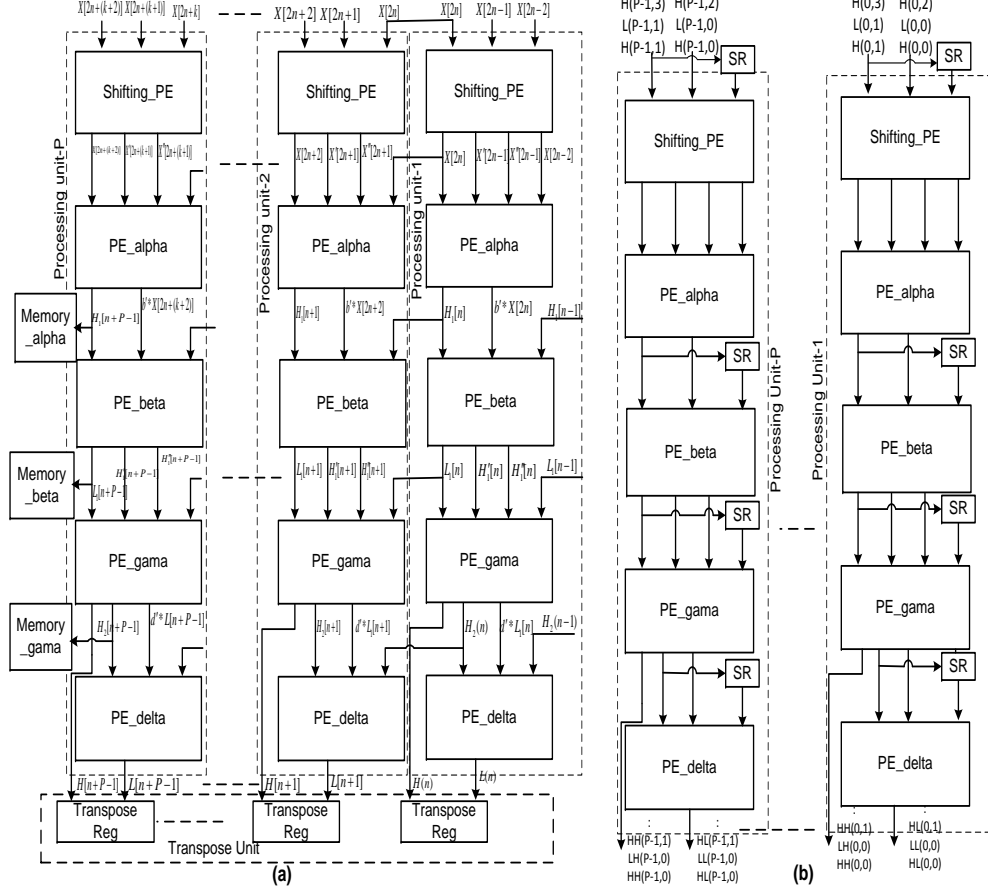


Figure 3. (a)Row Processor (b) Column Processor

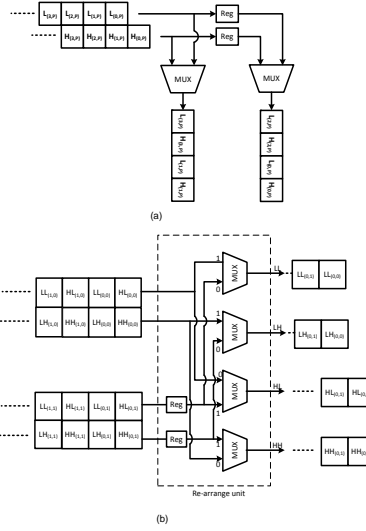


Figure 4. (a) Transpose Register (Ref:[17]) (b) Re-arrange Unit

for a row processor with  $P$  number of PUs.  $P = 2$  has been considered for our proposed design. For the first clock cycle, RP get the pixels from  $X(0, 0)$  to  $X(0, 2P)$  simultaneously. For the second clock RP gets the pixels from next row i.e.  $X(1, 0)$  to  $X(1, 2P)$ , the same procedure continues for each clock till it reaches the bottom row i.e.,  $X(N, 0)$  to  $X(N, 2P)$ . Then it goes to the next strip and RP get the pixels from  $X(0, 2P)$  to  $X(0, 4P)$  and it continues this procedure for entire image. Each PU consists of five pipeline stages and each pipeline stage is processed by one processing element (PE) as depicted in Fig. 2(b). First stage (shift\_PE) provide the partial results which is required at  $2^{nd}$  stage (PE\_alpha), likewise processing elements PE\_alpha to PE\_delta ( $2^{nd}$  stage to  $5^{th}$  stage) gives the partial results along with their original outputs. (e.g., consider the PE\_alpha of PU-1, it needs to provide output corresponding to eqn.(1) ( $H_1[n]$ ), along with  $H_1[n]$ , it also provides the partial output  $X'[2n]$  which is required for the PE\_beta). Structure of the PEs are given in the Fig. 2(b), it shows that multiplication is replaced with the shift and add technique. The original multiplication factor and the value through the shift and add circuit are noted in Table.I, it shows that variation between original and adopted one is extremely small. As shown in Fig. 2(b), time delay of shift\_PE is one  $T_a$  and remaining all PEs are having delay of  $2T_a$ . To reduce the CPD of PU, PEs from PE\_alpha to PE\_delta are divided in to two pipeline stages, and each pipeline stage has a delay of  $T_a$ , as a result CPD of PU is reduced to  $T_a$  and pipeline stages are increased to nine and is shown in Fig. 2(c). The outputs  $H_1[n + P - 1]$ ,  $L_1[n + P - 1]$ , and  $H_2[n + P - 1]$  corresponding to PE\_alpha and PE\_beta of last PU and PE\_gamma of last PU is saved in the memories Memory\_alpha, Memory\_beta and Memory\_gamma respectively, shown in Fig. 3(a). Those stored outputs are inputted for next subsequent columns of the same row. For a  $N \times N$  image rows is equivalent to  $N$ . So the size of the each memory is  $N \times 1$  words and total row memory to store these outputs is equals to  $3N$ . Output of each PU are under gone through a process of scaling before it producing the outputs H and L. These outputs are fed to the transposing unit. The transpose unit has  $P$  number of transpose registers (one for each PU). Fig. 4(a) shows the structure of transpose register, and it gives the two H and two L data alternatively to the column processor.

2) *Column Processor (CP)*: The structure of the column processor (CP) is shown in Fig. 3(b). To match with the throughput of RP, CP is also designed with two number of PUs in our architecture. Each transpose register produces a pair of H and L in an alternative order and are fed to the inputs of one PU of the CP. The partial results produced are consumed by the next PE after two clock cycles. As such, shift registers of length two are needed within the CP between each pipeline stages for caching the partial results



Table I  
ORIGINAL AND ADOPTED VALUES FOR MULTIPLICATION

PE	Original Multiplier Value	Multiplier value through shift and add
PE_alpha	$a'=-0.6305$	$a'=-0.6328$
PE_beta	$b'=11.90$	$b'=12$
PE_gama	$c'=-21.378$	$c'=-21.375$
PE_delta	$d'=2.55$	$d'=2.5625$

(except between 1<sup>st</sup> and 2<sup>nd</sup> pipeline stages). At the output of the CP, four sub-bands are generated in an interleaved pattern, *i.e.*  $(HL, HH), (LL, LH), (HL, HH), (LL, LH)$ , and so on. Outputs of the CP are fed to the re-arrange unit. Fig. 4(b) shows the architecture for re-arrange unit, and it provides the outputs in sub-band order *i.e.*  $LL, LH, HL$  and  $HH$  simultaneously, by using  $P$  registers and  $2P$  multiplexers. For multilevel decomposition, the same DWT core can be used in a folded architecture with an external frame buffer for the LL sub-band coefficients.

#### B. Architecture for Temporal Processor (TP)

Eqn.(8) shows that Haar wavelet transform depends on two adjacent pixels values (same pixel position of adjacent frames, for temporal processing). As soon as spatial processors are provide the 2-D DWT results, temporal processors starts processing on the spatial processor outputs (2-D DWT results) and produce the 3-D DWT results. Fig. 1 shows that there is no requirement of temporal buffer, due to the sub-band coefficients of two spatial processors are directly connected to the four temporal processors. But it has been designed with 3 pipeline stages, it require 6 pipeline registers for each TP. Same frequency sub-band of the distinct spatial processors are fed to the each temporal processor. *i.e.*  $LL, HL, LH$  and  $HH$  sub-bands of the spatial processor 1 and 2 are given as inputs to the temporal processor 1, 2, 3 and 4 respectively. Temporal processor apply 1-D Haar wavelet on sub-band coefficients, and provide the low frequency sub-band and high frequency sub-band as output. By combining all low frequency sub-bands and high frequency sub-bands of all temporal processors provide the 3-D DWT output in the form of L-Frame and H-Frame (2-D DWT by spatial processors and 1-D DWT by temporal processors).

### IV. IMPLEMENTATION RESULTS AND PERFORMANCE COMPARISON

The proposed 3-D DWT architecture has been described in Verilog HDL. A uniform word length of 14 bits has been maintained throughout the design. Simulation results have been verified by using Xilinx

Table II  
DEVICE UTILISATION SUMMARY OF THE PROPOSED ARCHITECTURE

Logic utilized	Used	Available	Utilization
Slice Registers	1958	106400	1%
Number of Slice LUTs	2852	53200	5%
Number of fully used LUT-FF pairs	1137	3673	30%
Number of Block RAM	3	140	2%

Table III  
COMPARISON OF PROPOSED 2-D DWT ARCHITECTURE WITH EXISTING ARCHITECTURES (FOR 1-LEVEL)

Parameter	Zhang [12]	Mohanty [13]	Darji [14]	Yusong [17]	Proposed
Multipliers	10	9P	10	10P	0
Adders	16	16P	16	16P	34P
Internal Memory	$4N+37$	$15P+5.5N$	$4N$	$24P+3N$	$60P+3N$
Critical path	$T_m$	$T_m + 2T_a$	$T_m$	$T_m + T_a$	$T_a$
Computation Time	$N^2/2$	$N^2/2P$	$N^2/2$	$N^2/2P$	$N^2/2P$
Throughput	$2/T_m$	$2P/T_m + 2T_a$	$2/T_m$	$2P/T_m + T_a$	$2P/T_a$

Table IV  
COMPARISON OF PROPOSED 3-D DWT ARCHITECTURE WITH EXISTING ARCHITECTURES (FOR 1-LEVEL)

Parameters	Weeks [19]	Taghavi [20]	A.Das [22]	Darji [24]	Proposed
Memory requirement	$6N^2+6l$	$5N^2$	$5N^2 + 5N$	$4N^2 + 10N$	$2*(3N+60P)+48$
Throughput/cycle	-	1 result	2 results	4 results	8 results
Computing time For 2 Frames	$2N^2 + 3l/2$	$6N^2$	$3N^2$	$3N^2$	$N^2/2P$
Latency	$2.5N^2 + 0.5l$	$4N^2$ cycles	$2N^2$ cycles	$3N^2/2$ cycles	21 cycles
Area	-	-	1825 slices	2490 slices	2852 slice LUTs
Operating Frequency	200 MHz (ASIC)	-	321 MHz (FPGA)	91.87 MHz (FPGA)	265 MHz (FPGA)
Multipliers	-	-	Nil	30	Nil
Adders	$6l$ MACs	-	78	48	168
Filter bank	$l$ -length	D-9/7	D-9/7	D-9/7	D-9/7 (2-D) + Haar (1-D)

Table V  
SYNTHESIS RESULTS (DESIGN VISION) COMPARISON OF PROPOSED 3-D DWT ARCHITECTURE WITH EXISTING

Parameters	Darji et al.,[24]	Proposed
Comb. Area	$61351 \mu m^2$	$526419 \mu m^2$
Non Comb. Area	$807223 \mu m^2$	$553078 \mu m^2$
Total Cell Area	$868574 \mu m^2$	$1079498 \mu m^2$
Operating Voltage	1.98 V	1.2 V
Total Dynamic Power	179.75 mW	38.56 mW
Cell Leakage Power	$46.87 \mu W$	4.86 mW

ISE simulator. We have simulated the Matlab model which is similar to the proposed 3-D DWT hardware architecture and verified the 3-D DWT coefficients. RTL simulation results have been found to exactly match the Matlab simulation results. The Verilog RTL code is synthesized using Xilinx ISE 14.2 tool and mapped to a Xilinx programmable device (FPGA) 7z020clg484 (zynq board) with speed grade of -3. Table II shows the device utilization summary of the proposed architecture and it operates with a maximum frequency of 265 MHz.

The proposed architecture has also been synthesized using SYNOPSYS design compiler with 90-nm technology CMOS standard cell library. It consumes 43.42 mW power and occupies an area equivalent to 231.45 K equivalent gate at frequency of 200 MHz.

#### A. Comparison

The performance comparison of the proposed 2-D and 3-D DWT architectures with other existing architectures is figure out in Tables III and IV respectively. The proposed 2-D processor requires zero multipliers,  $34P$  (Pis number of parallel PUs) adders,  $60P+3N$  internal memory. It has a critical path delay of  $T_a$  with a throughput of four outputs per cycle with  $N^2/2P$  computation cycles to process an image with size  $N \times N$ . When compared to recent 2-D DWT architecture developed by the Y.Hu et al. [17], CPD reduced to  $T_a$  from  $T_m + T_a$  with the cost of small increase in hardware resources.

Table IV shows the comparison of proposed 3-D DWT architecture with existing 3-D DWT architecture. It is found that, the proposed design has less memory requirement, High throughput, less computation time and minimal latency compared to [19], [20], [22], and [24]. Though the proposed 3-D DWT architecture has small disadvantage in area and frequency, when compared to [22], the proposed one has a great advantage in remaining all aspects.

Table V gives the comparison of synthesis results between the proposed 3-D DWT architecture and [24]. It seems to be proposed one occupying more cell area, but it included total on chip memory also, where as in [24] on chip memory is not included. Power consumption of the proposed 3-D architecture is very less compared to [24].

## V. CONCLUSIONS

In this paper, we have proposed memory efficient and high throughput architecture for lifting based 3-D DWT. The proposed architecture is implemented on 7z020clg484 FPGA target of zynq family, also

synthesized on Synopsys' design vision for ASIC implementation. An efficient design of 2-D spatial processor and 1-D temporal processor reduces the internal memory, latency, CPD and complexity of a control unit, and increases the throughput. When compared with the existing architectures the proposed scheme shows higher performance at the cost of slight increase in area. The proposed 3-D DWT architecture is capable of computing 60 UHD ( $3840 \times 2160$ ) frames in a second.

## REFERENCES

- [1] Q. Dai, X. Chen, and C. Lin, "A Novel VLSI Architecture for Multidimensional Discrete Wavelet Transform," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, No. 8, pp. 1105-1110, Aug. 2004.
- [2] C. Cheng and K. K. Parhi, "High-speed VLSI implementation of 2-D discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 393-403, Jan. 2008.
- [3] B. K. Mohanty and P. K. Meher, "Memory-Efficient High-Speed Convolution-based Generic Structure for Multilevel 2-D DWT," *IEEE Transactions on Circuits and Systems for Video Technology*, VOL. 23, NO. 2, pp. 353-363, Feb. 2013.
- [4] I. Daubechies and W. Sweledens, "Factoring wavelet transforms into lifting schemes," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 247-269, 1998.
- [5] C.T. Huang, P.C. Tseng, and L.-G. Chen, "Flipping structure: An efficient VLSI architecture for lifting-based discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 1080-1089, Apr. 2004.
- [6] C.-Y. Xiong, J.-W. Tian, and J. Liu, "A Note on Flipping Structure: An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform," *IEEE Transactions on Signal Processing*, Vol. 54, No. 5, pp. 1910-1916, MAY 2006
- [7] C.-T. Huang, P.-C. Tseng, and L.-G. Chen, "Analysis and VLSI architecture for 1-D and 2-D discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1575-1586, Apr. 2005.
- [8] C.-C. Cheng, C.-T. Huang, C.-Y. Ching, C.-J. Chung, and L.-G. Chen, "On-chip memory optimization scheme for VLSI implementation of line based two-dimensional discrete wavelet transform," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 7, pp. 814-822, Jul. 2007.
- [9] H.Y. Liao, M. K. Mandal, and B. F. Cockburn, "Efficient architectures for 1-D and 2-D lifting-based wavelet transforms," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1315-1326, May 2004.
- [10] B.F. Wu and C.F. Chung, "A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 12, pp. 1615-1628, Dec. 2005.
- [11] C.-Y. Xiong, J. Tian, and J. Liu, "Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 607-614, Mar. 2007.
- [12] W. Zhang, Z. Jiang, Z. Gao, and Y. Liu, "An efficient VLSI architecture for lifting-based discrete wavelet transform," *IEEE Transactions on Circuits and Systems-II: Express Briefs*, Vol. 59, No. 3, pp. 158-162, Mar. 2012.
- [13] B. K. Mohanty and P. K. Meher, "Memory Efficient Modular VLSI Architecture for High throughput and Low-Latency Implementation of Multilevel Lifting 2-D DWT," *IEEE Transactions on Signal Processing*, Vol. 59, No. 5, pp. 2072-2084, May 2011.
- [14] A. Darji, S. Agrawal, Ankit Oza, V. Sinha, A. Verma, S. N. Merchant and A. N. Chandorkar, "Dual-Scan Parallel Flipping Architecture for a Lifting-Based 2-D Discrete Wavelet Transform," *IEEE Transactions on Circuits and Systems-II: Express Briefs*, Vol. 61, No. 6, pp. 433-437, Jun. 2014.

- [15] B. K. Mohanty, A. Mahajan, and P. K. Meher, "Area and power efficient architecture for high-throughput implementation of lifting 2-D DWT," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 7, pp. 434-438, Jul. 2012.
- [16] Y. Hu and C. C. Jong, "A Memory-Efficient High-Throughput Architecture for Lifting-Based Multi-Level 2-D DWT," *IEEE Transactions on Signal Processing*, VOL. 61, NO. 20, pp.4975-4987, Oct. 15, 2013.
- [17] Y. Hu and C. C. Jong, "A Memory-Efficient Scalable Architecture for Lifting-Based Discrete Wavelet Transform," *IEEE Transactions on Circuits and Systems-II: Express Briefs*, VOL. 60, NO. 8, pp. 502-506, Aug. 2013.
- [18] J. Xu, Z.Xiong, S. Li, and Ya-Qin Zhang, "Memory-Constrained 3-D Wavelet Transform for Video Coding Without Boundary Effects," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, No. 9, pp. 812-818, Sep. 2002.
- [19] M. Weeks and M. A. Bayoumi, "Three-Dimensional Discrete Wavelet Transform Architectures," *IEEE Transactions on Signal Processing*, Vol. 50, No. 8, pp.2050-2063, Aug. 2002.
- [20] Z. Taghavi and S. kasaei, "A memory efficient algorithm for multidimensional wavelet transform based on lifting," in *Proc. IEEE Int. Conf. Acoust Speech Signal Process. (ICASSP)*, vol. 6, pp. 401-404, 2003.
- [21] Q. Dai, X. Chen, and C. Lin, "Novel VLSI architecture for multidimensional discrete wavelet transform," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 8, pp. 1105-1110, Aug. 2004.
- [22] A. Das, A. Hazra, and S. Banerjee, "An Efficient Architecture for 3-D Discrete Wavelet Transform," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 20, NO. 2, pp. 286-296, Feb. 2010.
- [23] B. K. Mohanty and P. K. Meher, "Memory-Efficient Architecture for 3-D DWT Using Overlapped Grouping of Frames," *IEEE Transactions on Signal Processing*, Vol. 59, No. 11, pp.5605-5616, Nov. 2011.
- [24] A. Darji, S. Shukla, S. N. Merchant and A. N. Chandorkar, "Hardware Efficient VLSI Architecture for 3-D Discrete Wavelet Transform," *Proc. of 27<sup>th</sup> Int. Conf. on VLSI Design and 13<sup>th</sup> Int. Conf. on Embedded Systems* pp. 348-352, 5-9 Jan. 2014.
- [25] W.Sweldens, "The Lifting Scheme: a Construction of Second Generation of Wavelets," *SIAM Journal on Mathematical Analysis*, Vol.29 No.2, pp. 511-546, 1998.